CLAIMS

What is claimed is:

1. A method comprising:

assigning an identification number (ID) to each of a plurality of micro-operations (uops) to identify a branch path to which the uop belongs;

determining whether one or more branches are predicted correctly;

determining which of the one or more branch paths are dependent on a mispredicted branch; and

determining whether one or more of the plurality of uops belong to a branch path that is dependent on the mispredicted branch based on their assigned IDs.

2. The method of claim 1, further comprising retiring a uop that belongs to a branch path dependent on a mispredicted branch.

3. The method of claim 1, further comprising assigning each of the plurality of uops a sequence number.

4. The method of claim 3, further comprising storing the sequence number of an oldest valid uop in each branch path.

5. The method of claim 4, further comprising comparing the sequence number of a uop to the sequence number of the oldest valid uop in a same branch path.

6. The method of claim 1, further comprising maintaining a list of available IDs.

7. The method of claim 6, wherein assigning an ID to each of a plurality of uops to identify a branch path to which the uop belongs comprises assigning by an allocator an ID for each of the plurality of uops from the list of available IDs.

8. The method of claim 7, further comprising stalling the allocator if there is no available ID to be assigned.

9. The method of claim 7, further comprising placing an ID on the list of available IDs when all uops that have been assigned that ID have been retired.

10. An apparatus comprising:

an allocator to assign a plurality of micro-operations (uops) identification numbers (IDs), each ID to identify a branch path to which the uop belongs;

a jump unit coupled to the allocator to determine whether branches are predicted correctly; and

an execution unit coupled to the jump unit to determine which uops belong to a branch path that is dependent on a mispredicted branch based on their assigned IDs.

11. The apparatus of claim 10, further comprising a retire unit coupled to the jump unit to retire uops that are related to a mispredicted branch.

12. The apparatus of claim 11, wherein the allocator to further maintain a list of available IDs and assign an ID for each branch from the list of IDs.

13. The apparatus of claim 12, wherein the retire unit to further place an ID on the list of available IDs when all uops that have been assigned that ID have been retired.

14. The apparatus of claim 10, wherein the allocator to further assign each of the plurality of uops a sequence number.

15. The apparatus of claim 14, wherein the jump unit to further store the sequence number of the oldest valid uop in each branch path.

16. The apparatus of claim 15, wherein the execution unit to further compare the sequence number of a uop to the sequence number of an oldest valid uop in a same branch path.

17. The apparatus of claim 10, further comprising an instruction fetch unit coupled to the allocator to fetch a next instruction based on a next instruction pointer.

18. The apparatus of claim 17, further comprising an instruction decode unit coupled to the instruction fetch unit to decode the fetched instructions.

19. A system comprising:

an input/output (I/O) controller; and

a processor coupled to the I/O controller, the processor including:

an allocator to assign micro-operations (uops) identification numbers (IDs), each ID to identify a branch path to which the uop belongs;

a jump unit coupled to the allocator to determine whether branches are predicted correctly; and

an execution unit coupled to the jump unit to determine which uops belong to a branch path that is dependent on a mispredicted branch based on their assigned IDs.

20. The system of claim 19, wherein the processor further comprises a retire unit coupled to the jump unit to retire uops that are related to a mispredicted branch.

21. The system of claim 19, wherein the processor further comprises an instruction fetch unit coupled to the allocator to fetch a next instruction based on a next instruction pointer.

22. The system of claim 21, wherein the processor further comprises an instruction decode unit coupled to the instruction fetch unit to decode the fetched instructions.